Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 54 (2015) 422 – 430

Procedia Computer Science

Eleventh International Multi-Conference on Information Processing-2015 (IMCIP-2015)

A Privacy Preserved Data Mining Approach Based on k-Partite Graph Theory

T. Pranav Bhat, C. Karthik* and K. Chandrasekaran

Department of Computer Science and Engineering, NITK Surathkal 575 025, Karnataka, India

Abstract

Traditional approaches to data mining may perform well on extraction of information necessary to build a classification rule useful for further categorisation in supervised classification learning problems. However most of the approaches require fail to hide the identity of the subject to whom the data pertains to, and this can cause a big privacy breach. This document addresses this issue by the use of a graph theoretical approach based on k-partitioning of graphs, which paves way to creation of a complex decision tree classifier, organised in a prioritised hierarchy. Experimental results and analytical treatment to justify the correctness of the approach are also included.

Peer-review under responsibility of organizing committee of the Eleventh International Multi-Conference on Information Processing-2015 (IMCIP-2015)

Keywords: Data mining; Graph theory; K-partite; Privacy; Security.

1. Introduction

Information extraction from a given data-repository to determine the behaviour of a particular system, or to determine the predictive outcome of a particular problem statement for the case of an unknown condition or input forms an application of wide horizon in easing the life of human beings, by its penetration into domains ranging from e-commerce to healthcare. Supervised learning algorithms have been widely employed in prediction problems to forecast the outcome of a tweaked problem from an underlying data reflecting the actual outcomes of similar problems.

A major concern that arises out of the above techniques of data repositories for data mining using supervised learning techniques for building of classification rules is the privacy and confidentiality of the information, especially in guarding the identity of the subjects to whom the information pertains to. Various privacy issues could arise due of the mining of such sensitive personal data, and misuse of the data by breach of privacy can cause legal and ethical issues beyond the domain of data mining

Privacy Preserved Data Mining is a new hype which has entered the market and which claims to take care of this particular issue. The goal of privacy preserving data mining is to develop data mining methods without increasing the risk of misuse of the data used to generate those methods. Literature cites a large number of methods, most of which use some form of transformation on the original data to ensure privacy preservation, called key interchange mapping methods, but these methods are quite complex and compute and memory intensive, thus leading to limited

*Corresponding author. Tel.: +91 9035219859.
E-mail address: karthikiyer2000@gmail.com

Peer-review under responsibility of organizing committee of the Eleventh International Multi-Conference on Information Processing-2015 (IMCIP-2015)
doi:10.1016/j.procs.2015.06.049

usage of these methods. This document suggests an alternative approach to privacy preservation. This method leads to identification of two categories of attributes – *key – attributes*, which directly reveal the identity of an individual with minimal or single operations on them and *quasi – identifier – attributes*, which identify an individual through certain data mining operations, primarily due to the existence of attribute dependencies. The proposed technique harnesses these vulnerabilities in privacy-unpreserved dataset and strives to eliminate these, by using simple principles from the theory of *k-partite graphs*, and builds classifiers from these privacy preserved data-subsets, which then will be grouped based on decision tree root priority approaches, to form a privacy preserving complex classifier or classification rule for test sets.

We organise the paper through 7 sections in the following fashion. Section 2 discusses related work and earlier contributions cited in literature in the domain of privacy preserved data mining. Section 3 details the methodology proposed with the required analytical justifications wherever necessary. Section 4 illustrates the approach and justifies it by the use of an experimental setup, and also analysis of results, followed by conclusion in Section 6.

2. Literature Survey

A lot of work has gone into tackling the issues of data related security. One of the recent issues is the privacy preservation of users and individuals while mining through data. The work by Agarwal and Srikant[1] on PPDM is one of the initial works to address this issue. In their paper they have built a decision tree using training data whose distribution was scattered and still obtained comparable classification accuracy results. In[2,3] the authors have given a detailed description about the *k* anonymization and randomization techniques of PPDM and also addressed the issues and the areas of application for PPDM. In[4] a detailed study has been given of topics such as attribute relations, use of technology for privacy enhancement. They have done this through a survey of data mining related privacy for two methods-randomization and secure multiparty computation. In[5] the authors have proposed a two tier method by which the medical data can be safely mined with increased privacy. The two tiers are horizontal data separation and vertical data separation. A similar work was done in[6] where the authors decided the level of anonymization of attributes based on their sensitivity. In[7] the authors have proposed an enhancement of the *k* anonymity method for privacy preservation.

3. Methodology

The principle of enabling privacy preservation in a dataset under use mainly concerns with the identification of the vulnerabilities or faults in the existing data-mining methodology, or in the existing set of steps involved in the information retrieval process using data mining techniques based on supervised learning concepts. More specifically privacy breach and information misuse can be avoided by eliminating direct or indirect extraction of information pertaining to the subjects of the particular information, especially when the information is quite sensitive as in the cases of healthcare data. This drives home the idea that identity related attributes need to be eliminated or anonymised for privacy preservation, which means that the key attributes can be removed and the quasi-attributes can be played around with, which is the major principle driving privacy preservation in the proposed approach.

The initial cleansing and formatting process is performed on the target data, which is followed by the graph theoretical privacy preservation proposed to generate privacy preserving sub-classifiers, which are then integrated in a decision tree root identification hierarchy methodology proposed by Quinlan *et al.*[8], followed by the classification. This forms the complex privacy preserved data mining setup.

3.1 Assumptions

1. Applicability to supervised learning approaches by the existence of a labelled training dataset.
2. Problem is a binary classification problem (or also called a *concept*), where the output is either true or false (this condition can be relaxed since the method works for multi-class classifications as well.
3. Data stored as a relational database. Data stored in the form of semi-structured data, in the form of XML sheets or NO-SQL databases may need to be transformed into Relational databases and then this method applied.

4. Input attribute values are also partitioned into classes.
5. Attribute dependencies and rules of inference which forms the background knowledge for the problem specific domain, are already provided by the domain expert.

3.2 Proposed PPDM procedure

1. **Obtaining the Dataset:** A suitable dataset is chosen from popular repositories like UCI, Kaggle, Reuters and so on, or the dataset self generated, based on the given domain knowledge.
2. **Dataset Cleaning:** Usually required for self collected or spawned dataset, it involves error handling of the dataset by attribute sufficiency determination and by identification and removal of the inconsistent examples from the training set. Literature cites various methods for data cleansing. However datasets from standard repositories usually do not require this step, since the data is usually clean.
3. **Privacy preservation by elimination of direct and indirect identities:** This is the primary step involved in the elimination of identity information related to the subject to whom the data pertains to. The steps involved in this are,
   (a) *Determining and Removing Candidate key attribute(s)* - Involves identification of those attributes which have values unique to each data object or record, and not a form of a floating point data (since many floating point numbers can exist between two floating point numbers.). These attributes can be removed directly since it may not be a case that these will contribute to the classifier building process.This residues with *quasi – attributes*.
   (b) *Dependency Elimination and Database Normalisation:* Regular database normalisation is performed, but the attribute partitions are determined by theory of *k-partite graphs*, where the nodes represent the attributes and the edges represent the dependencies between them, and then Algorithm 1 is applied on that. (Fig. 1 as example) A sample partitioning algorithm can be as follows:
   The idea behind this technique is that dependencies are creators of quasi-attributes since they potentially indicate the identity of the record subject and hence breaking these dependencies by partitioning can ensure privacy preservation, without loss of information, since it is a specific form of normalisation.
4. **Tournament Selection:** A random subset of the original dataset is horizontally chosen and repeatedly chosen, a technique called tournament selection , to create ten to fifteen data subsets depending on the problem domain.
   The generated data subsets are then subject to selective averaging involving choosing an attribute at random, and averaging a subset of the values of this attribute into the other subset, to ensure another level of privacy preservation. These data subsets are then vertically split to create k data subsets from each subset.
5. **Classifier Generation and Classifier Compounding:** The obtained $10k$ data subsets are then fed into a suitable decision tree generator module in a suitable programming platform, and $10k$ sub-classifiers created. *Classifier Compounding* involves combining of decision sub-classifiers generated appropriately based on domain

Algorithm 1 Partitioning Algorithm (Graph 'G' = (V , E) of attribute dependencies)

```
1: G_set = G
2: repeat{
3: G_old = G_set
4: ∀x ∈ G_set{
5: (tcheck, G_temp) = CheckAndPartition(x)
6: check = check&tcheck
7: if (tcheck == True) then
8:     G_set = {G_set − x} U G_temp
9: else
10:    let G_temp = {v1, v2}
11:    Add a new vertex v_x to subgraph x
12:    Add edges e_i = (v1, v_x) and e_j = (v_x, v2)to x
13:    Remove edge e_k = (v1, v2) from x
14: }
15: }while(G_old ! = G_set)
16: return the vertex subsets of G_old
```

Algorithm 1. Partitioning algorithm (Graph 'G' $= (V, E)$ of attribute dependencies).

Algorithm 2 CheckAndPartition(x)

```
∀v ∈ vertex set V of G {
keep v uncoloured or − 14
}
Let source be a random node s
Mark src with red
Add src to set R
Consider a Queue Q
Push Source src into Q
while (q is not empty){
Dequeue node u from the Queue Q
for every vertex v adjacent to u{
if v is uncoloured then
    if u is red then
        Mark v with blue
        Add v to set B
    else
        Mark v with red
        Add v to set R
    Push v into Q
else if v and u are of the same colour then
    return{false, {u, v}}
return{true, {R, B}}
```

Algorithm 2. CheckAndPartition(x).

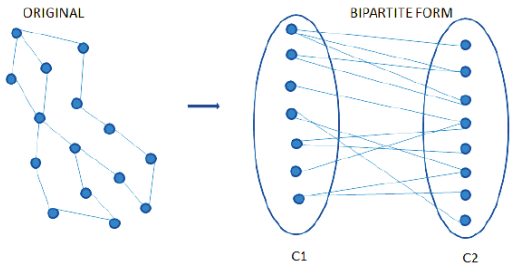

Fig. 1. Graph conversion to bipartite form by algorithm 1.

knowledge and the sub-classifier priority. The sub-classifier priority is defined as the average score of the attributes involved in the sub-classifier, and the attribute scores are based on the attribute gains and Information weights of the attributes as defined by Quinlan *et al.*[13], indicated by equations 1, 2, 3, 4, 5 and additional proposed equations 6, 7 and 8, as given below,

Let the total number of positive instances as p and number of negative instances as n.

The total weight of a decision tree for these samples is given by $I(p,n)$, defined by,

$$I(p,n) = -\frac{p}{p+n}\log_2\frac{p}{p+n} - \frac{n}{p+n}\log_2\frac{n}{p+n} \tag{1}$$

Considering an attribute A_i with m outcomes $o_1, o_2, \ldots, o_m$, with p_x = number of positive instances with outcome o_x and n_x = number of negative instances with outcome o_x, the following parameters can be defined as,

(a) Expectation of A_i, which is the weight hanging below A_i if chosen as the root is, given by,

$$E(A_i) = \sum_{j=1}^{m}\left(\frac{p_j+n_j}{p+n}\text{Weight}(\text{Subtree}(A_i))\right) \tag{2}$$

$$\text{implies, } E(A_i) = \sum_{j=1}^{m}\frac{p_j+n_j}{p+n}I(p_j,n_j) \tag{3}$$

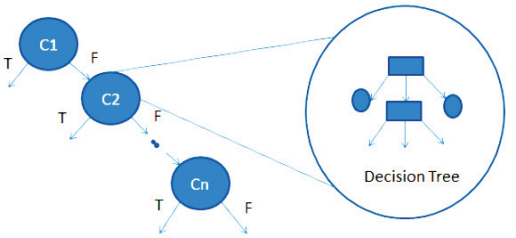

Fig. 2. A compound decision tree classifier (sub-classifier in inset) – false cascading.

(b) Gain of A_i, given by,

$$G(A_i) = I(p,n) - E(A_i) \tag{4}$$

(c) Information Value of A_i given by,

$$IV(A_i) = -\sum_{j=1}^{m} \frac{p_j + n_j}{p+n} \log_2 \frac{p_j + n_j}{p+n} \tag{5}$$

(d) Weight of A_i, given by,

$$Wt(A_i) = G(A_i)/IV(A_i) \tag{6}$$

The for every classifier c_j in the group $c_1, c_2, \ldots, c_k$ of a particular data subset, we find the score of the classifier, as

$$S(c_j) = \frac{\sum_{j=1}^{m} Wt(A_i)}{m} \tag{7}$$

where c_j contains m attributes $A_1, A_2, \ldots, A_m$.

Then we place the classifiers in the complex classifier tree in an order of decreasing $S(c_j)$. That is we store it as (Fig. 2)

$$S(c_1) >= S(c_2) >= \cdots >= S(c_k) \tag{8}$$

The crux of the problem also lies in choosing the appropriate outcome as the exit condition (illustrated by T in Fig. 2), and this is generally decided by the domain knowledge of the system.

Thus the above step creates a set of 10 compound-classifiers, each having different efficiencies, but the same attributes, but different internal structures of the sub-classifiers which are nodes. This process is called as *Classifier Compounding*.

The final result will be an average over the classification efficiencies of the 10 compound classifiers when run using our dataset.

4. Experimental Setup

Table 1 defines the setup of the experiment in terms of the parameters being used in it.

1. Candidate Key elimination involved the removal of attributes SSN Number and Mobile Number, as per the candidate key selection criteria.
2. The dependencies was as found as in Fig. 1, where each node represents one of the 15 attributes, which was then split into two partitions $C1$ and $C2$ as shown. Hence $k = 2$ for 2-*partite* graph was formed.
3. The values of the different parameters to decide the sub-classifier score was found out to be as shown in Table 2. (Note: Dependencies are dummy dependencies, and may not pertain to actual relationships).
4. *Tournament Selection:* Data set with 900 records and 15 attributes, converted to 10 sets of 15 attributes each via tournament selection, with the following procedure,

Table 1. Experimental setup.

Parameter	Value
Problem Domain	German Student Loan Sanction Prediction
Dataset Source	UCI Repository
Programming Platform	Python 2.7.3
Programming Module	Scikit Learn (sklearn-0.16b)
Dataset Size	1000
Training Set Size	900
Validation Set Size	100
Number of Attributes	15
Attribute List	SSN-Number
	SSN Number(*Candidate Key*)
	Mobile Number(*Candidate Key*)
	Loan duration
	Credit history/Credit rating
	Loan purpose
	Loan Amount Category
	Savings A/c status
	Present Employment Status
	Sex and Marital
	Guarantor Status
	Property Status
	Age Group
	Other Instalment
	Housing Status
	Job quality
	Number of Dependants
	Foreign Worker
Classifier Output	True/False

Table 2. Attribute weights, assuming same order as in table 1.

Attribute	$E(A)$	$G(A)$	$IV(A)$	$wt(A)$
Loan duration	0.8536	0.0276	1.7655	0.0156
Credit history/Credit rating	0.8376	0.0436	1.7118	0.0254
Loan purpose	0.8564	0.0248	2.5975	0.0095
Loan Amount Category	0.8651	0.0161	1.7198	0.0093
Savings A/c status	0.8531	0.0281	1.6877	0.0166
Present Employment Status	0.8681	0.0131	2.1551	0.006
Sex and Marital	0.8744	0.0068	1.5321	0.0044
Guarantor Status	0.8764	0.0047	0.5384	0.0089
Property Status	0.8643	0.0169	1.9477	0.0087
Age Group	0.8696	0.0115	1.3311	0.0087
Other Instalment	0.8724	0.0088	0.8447	0.0105
Housing Status	0.8685	0.0127	1.139	0.0111
Job quality	0.8799	0.0013	1.4134	0.0009
Number of Dependents	0.8812	0.0	0.6222	0.0
Foreign Worker	0.8754	0.0058	0.2283	0.0254

(a) From the dataset D containing 900 records, random datasets $D_1, D_2, \ldots, D_{10}$, each containing 400 records randomly chosen from D are created.
(b) Random attribute A_i chosen from the 15 attributes $A_1, A_2, \ldots, A_9$ present in D, $D_1, D_2, \ldots, D_{10}$. (Separate attribute A_i for each data subset D_k)
(c) For this attribute A_i (different in each dataset D_j), 250 random records are chosen, selectively averaged by copying their average to the remaining 150 records.
(d) For every set D_i, split it into k partitions vertically based on the bipartite graph partitions. In this example, it creates 2 partitions from each D_i, thus creating a total of 20 data subsets in two categories (10 in each category), to create 10 complex – classifiers.

Table 3. Sub-classifier scores by equation 7.

Sub-classifier	# of attributes	Weight
1	7(first 7)	0.01243
2	8(last 8)	0.009275

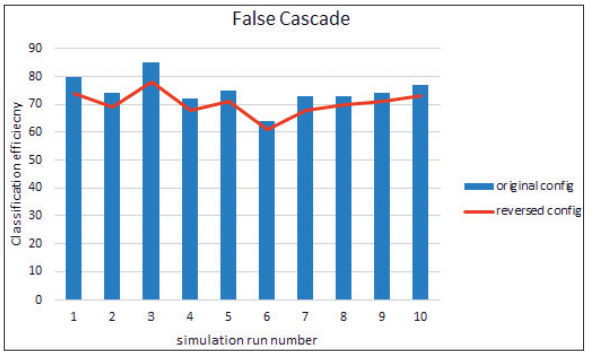

Fig. 3. Comparison of reversed and actual classifier configuration – false cascading (y axis shows the % accuracy).

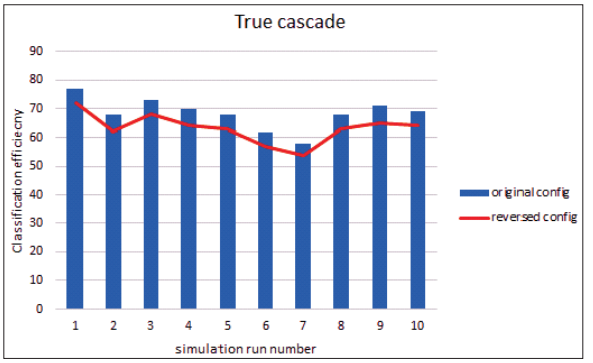

Fig. 4. Comparison of reversed and actual classifier configuration – true cascading (y axis shows the % accuracy).

5. Above 10 groups are then organised in a 2 level complex decision tree with the top node being the higher priority classifier with greater S value, and we use *True* as the exit branch (based on domain knowledge, but here verified experimentally as well)

Based on those values, we get the classifier scores as in Table 3,

5. Results and Analysis

Here are the results of the efficiencies, The graph shown in Fig. 3 represents the relative efficiency of two possible configurations of a compounded decision tree i.e. the better sub classifier at root and the lower classifier at the leaf and vice versa. The cascading followed here was a false cascading method wherein the traversal to the leaf node took place root sub classifier gave false. As it can be seen that the original configuration outperformed the reversed one for all the training and testing datasets for the given setup.

The graph in Fig. 4 shows the relative efficiency for original and reversed configurations for the case of false cascading. In spite of the change in the setup, the original classification still outperformed the reversed configuration for all the training and testing sets.

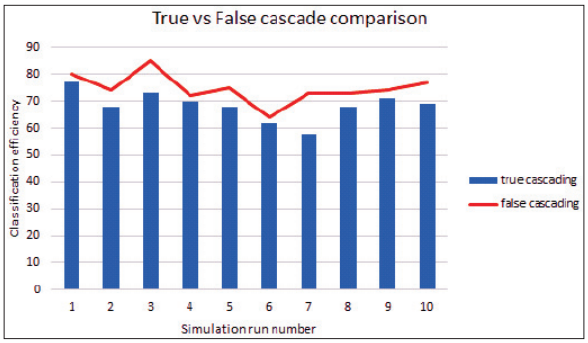

Fig. 5. Comparing false and true cascading (example specific) (y axis shows the % accuracy).

The graph in Fig. 5 is a comparison between the relative efficiency of false and true cascading (only for the original configurations as they outperformed the reversed configuration in both the cases). The result shows that false cascading method is more efficient compared to true cascading. In fact for one of the training – testing set the classification of false cascading went as high as 87%.

We enlist the following analysis from the dataset experiments which we ran,

- Our method gives the required efficiency in a very simple method. The average efficiency was found to be about 73%
- Our method of determining the order of the classifiers is considerably better
- More domain knowledge, better privacy

6. Conclusion

A newer method of privacy preservation of the dataset for the record subjects was proposed when used in the cases of data mining applications, especially in application related to medicine, military and finance, since confidentiality is a primary requirement here. This method is based on the removal of domain knowledge based attribute dependencies by their representation as a graph followed be k-partite partitioning, since these dependencies may reveal back the original identity of the person. In this context, mechanisms related to creation of complex classifier, tournament selection of the training set, root determination in the complex classifier mathematically by weighted averaging, were predicted and validated the results through application on a loan grant prediction domain, which justified the results experimentally.

We will be working on improving the drawbacks which we had discussed and speculated this improve the flexibility of the same. Further, we will be also trying to incorporate more training sets on which this experiment will validate the positivity of our outcome.

References

[1] R. Agrawal and R. Srikant, Privacy Preserving Data Mining, In *ACM SIGMOD International Conference on Management of data*, (2000).
[2] C. C. Aggarwal and P. S. Yu, A Course in Number Theory, *Privacy Preserving Data Mining: Models and Algorithms*, (2010).
[3] P. WANG, Survey on Privacy Preserving Data Mining, *International Journal of Digital Content Technology and its Applications*, vol. 4(9), (2010).
[4] J. Vaidya and C. Clifton, Privacy – Preserving Data Mining: Why, How, and when, (2007).
[5] G. Kou, Y. Peng, Y. Shi and Z. Chen, Privacy – Preserving Data Mining of Medical Fata using Data Separation Based Techniques, *Data Science Journal*, vol. 6, (2007).
[6] B. Abad and K. S.A, A Novel Approach for Privacy Preserving in Medical Data Mining using Sensitivity Based Anonymity, *International Journal of Computer Applications (0975–8887)*, vol. 42, (2012).
[7] R. Wong, J. Li, A. Fu and K. Wang, (,k)-Anonymity: An Enhanced k-Anonymity Model for Privacy Preserving Data Publishing, In *KDD*, Japan, pp. 754–759, (2006).
[8] J. R. Quinlan, Induction of Decision Trees, *Machine Learning*, vol. 1, pp. 569–571, (1986).
[9] X. Wu, *et al.*, Information Security in Big Data: Privacy and Data Mining, (2011).

[10] M. Paryasto, A. Alamsyah, B. Rahardjo and Kuspriyanto, Big – Data Security Management Issues, In *2nd International Conference on Information and Communication Technology*, (2014). .
[11] D. Niewolny, How the Internet of Things is Revolutionizing Healthcare, *Tech. Rep., FreeScale Technologies (Whitepaper)*, (2010).
[12] J. T. Kim, Privacy and Security Issues for Healthcare System with Embedded rfid System on Internet of Things, *Advanced Science and Technology Letters*, vol. 72, pp. 109–112, (2014).
[13] Will the Internet of Things Analytics Revolutionize the Healthcare Industry? *Tech. Rep. Saviance Technologies*, (2009).
[14] Tripathy, Animesh and M. Pradhan, A Novel Framework for Preserving Privacy of Data using Correlation Analysis, In *International Conference on Advances in Computing Communications and Informatics – ICACCI 12*, (2012).